IMPORT LIBRARIES

Import libraries for:

• loading embeddings

• handling matrices

• visualization

• t-SNE reduction

```
import sys
if 'gensim' not in sys.modules:
  !pip install gensim

import gensim.downloader as api          # Load pre-trained embeddings
import numpy as np                       # Matrix & numerical operations
import matplotlib.pyplot as plt          # Visualization
from sklearn.manifold import TSNE        # Dimensionality reduction
```

Show hidden output

LOAD EMBEDDING MODELS

• load pre-trained embeddings

• print vocabulary size

• display one example vector

```
# Load pre-trained Word2Vec (Google News 300-dimensional vectors)
model = api.load("word2vec-google-news-300")

# Print vocabulary size
print("Vocabulary size:", len(model))

# Display one example vector
example_word = "computer"
print("Vector for 'computer':")
print(model[example_word])
```

Show hidden output

SELECT WORD LIST

• choose 30–50 meaningful words

• store them in a list

• extract corresponding vectors

```
words = [
    # Animals
    "dog", "cat", "lion", "tiger", "wolf", "elephant", "horse",

    # Cities
    "Paris", "London", "Berlin", "Tokyo", "Beijing", "New_York", "Rome",

    # Technology
    "computer", "laptop", "keyboard", "mouse", "internet", "software", "hardware",

    # Fruits
    "apple", "banana", "orange", "grape", "mango", "pineapple", "peach",

    # Vehicles
    "car", "bus", "train", "airplane", "bicycle", "motorcycle"
]

# Extract vectors
vectors = np.array([model[word] for word in words])
```

APPLY t-SNE

• run t-SNE to reduce vectors from (e.g., 100/300 dims) → 2 dimensions

• store resulting coordinates

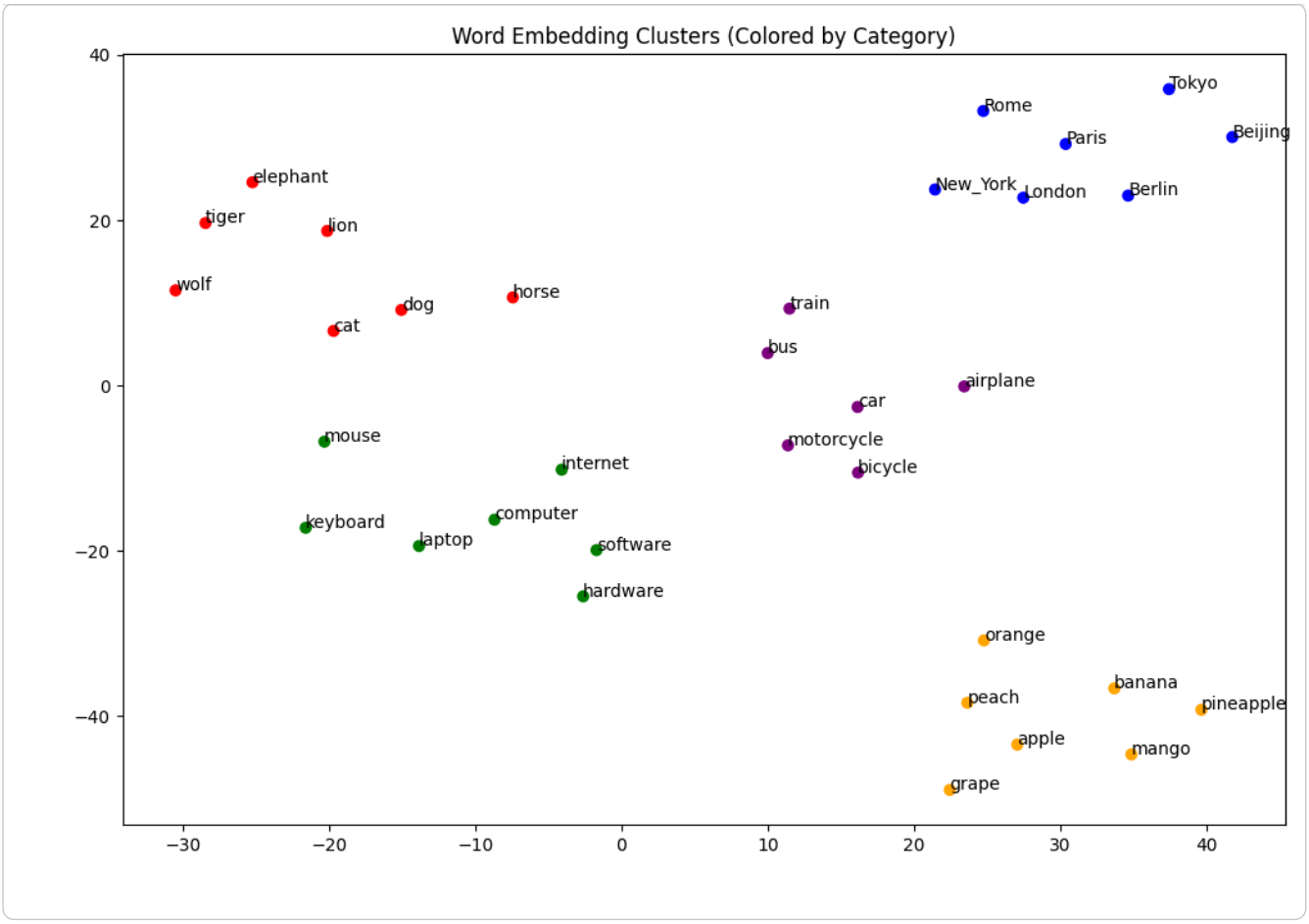
```
tsne = TSNE(n_components=2, random_state=42, perplexity=10)
reduced_vectors = tsne.fit_transform(vectors)
```

PLOT VISUALISATION

- create scatter plot
- annotate each point with word label
- highlight visible clusters

```
plt.figure(figsize=(12, 8))

x = reduced_vectors[:, 0]
y = reduced_vectors[:, 1]

plt.scatter(x, y)

# Annotate each word
for i, word in enumerate(words):
    plt.annotate(word, (x[i], y[i]))

plt.title("Word Embedding Visualization (t-SNE)")
plt.show()
```


```
#COLOR BY CATEGORY
colors = (
    ["red"] * 7 +        # Animals
    ["blue"] * 7 +       # Cities
    ["green"] * 7 +      # Technology
    ["orange"] * 7 +     # Fruits
    ["purple"] * 6       # Vehicles
)

plt.figure(figsize=(12, 8))
plt.scatter(x, y, c=colors)

for i, word in enumerate(words):
    plt.annotate(word, (x[i], y[i]))

plt.title("Word Embedding Clusters (Colored by Category)")
plt.show()
```


Word Embedding Clusters (Colored by Category)
Tokyo
Rome
Paris
Beijing
elephant
New_York
London
Berlin
tiger
lion
wolf
horse
dog
train
cat
bus
airplane
car
mouse
motorcycle
internet
bicycle
keyboard
computer
laptop
software
hardware
orange
banana
peach
pineapple
apple
mango
grape
40
20
0
−20
−40
−30
−20
−10
0
10
20
30
40